



Atmospheric Infrared Sounder (AIRS) Project

AIRS TLSCF Data System (TDS) Data Access Service

Quentin Sun

Quentin.Sun@jpl.nasa.gov

Rick Borgen, Albert Chang, Joe donhauser, Rick Forrsiter,
Quyen Nguyen, Frank Sun, Dean Ting

AIRS TLSCF Data System (TDS)

Outline



- Why TDS?
- Access Authorization
- Internal Data Organization
- Data Access Methods:
 - Directory Access
 - Catalog Access
- Catalog Access Examples

AIRS TLSCF Data System (TDS)



Why TDS?

- **Daily Data volume** is one of the main drivers
- Estimations are based on worst case 1x processing analysis for the first year of Mission Operations

Daily	Receive from GDAAC	Produce at TLSCF	(Total) Archive at TLSCF
Number of Files	2,450	3,610	6,060
File Size (Gbyte)	18	57	75

AIRS TLSCF Data System (TDS)

Access Authorization



- TDS Data Access is provided only to members of the AIRS Science Validation Team
- Each authorized user must acquire a **login account** for one of the TDS client computers (e.g., alpha.jpl.nasa.gov)
- Once log on, data users have **full access** to **all** science data products stored in the TDS

AIRS TLSCF Data System (TDS)

Internal Data Organization



- Data are organized into six logical groups called collections in TDS:
 - The *level0* collection
 - The *tlscf* collection
 - The *gdaac* collection
 - The *test* collection
 - The *sim* collection
 - The *correl* collection
- Collection type is one of the major metadata keywords used by the TDS catalog.

AIRS TLSCF Data System (TDS)

Directory Access



- The root of the TDS directory tree is always [/dom/files/airs/ops](#), regardless of which TDS client machine you log on.
- Below the root node, there are four sub-directory trees:
 - *airs* (i.e., [/dom/files/airs/ops/airs](#))
 - *test* (i.e., [/dom/files/airs/ops/test](#))
 - *sim* (i.e., [/dom/files/airs/ops/sim](#))
 - *correl* (i.e., [/dom/files/airs/ops/correl](#))



AIRS TLSCF Data System (TDS)

Directory Access – The *airs* Sub-tree

- The *airs* sub-tree contains three (3) branches identified by the **level0**, **tlscf** and **gdacc** collection names.
- Below each branch, data files are organized by *year, month, day* (day for granule file only) and *Earth Science Data Type (ESDT)*, respectively.
- For examples:
 - /dom/files/airs/ops/airs/**level0**/2001/06/*air10sci*/filename
 - /dom/files/airs/ops/airs/**level0**/2001/06/*airb0cap*/filename
 - /dom/files/airs/ops/airs/**tlscf**/2001/06/01/*airx2ret*/filename
 - /dom/files/airs/ops/airs/**tlscf**/2001/06/01/*airx2sup*/filename
 - /dom/files/airs/ops/airs/**gdacc**/2001/06/15/*airx2ret*/filename

AIRS TLSCF Data System (TDS)

Directory Access – The *correl* Sub-tree



- The *correl* sub-tree currently contains two (2) branches, namely, **ncep** and **arm**; additional branches will be configured as more correlative data become available.
- Below these branches, data files are organized by *year*, *month* and *ESDT*, respectively.
- For examples:
 - /dom/files/airs/ops/correl/**ncep**/2001/06/avi3_anh/filename
 - /dom/files/airs/ops/correl/**ncep**/2001/06/avi6_anh/filename
 - /dom/files/airs/ops/correl/**ncep**/2001/06/avi9_anh/filename
 - /dom/files/airs/ops/correl/**arm**/2001/05/airx2cac/filename
 - /dom/files/airs/ops/correl/**arm**/2001/04/airx2lac/filename



Directory Access – The *sim* and *test* Sub-tree

- The *sim* and *test* sub-trees contains sub-directories that are created in real-time, as needed.
- These subdirectory names must uniquely identify the simulation or test activity.
- Below the dynamically created directory, data files are organized by *year*, *month*, *day* (day for granule file only) and *ESDT*, respectively.
- For examples:
 - /dom/files/airs/ops/test/**XXX**/2001/06/01/airx2ret/filename
 - /dom/files/airs/ops/sim/**YYY**/2001/06/01/airx2ret/filenameWhere **XXX** and **YYY** could be **PGE2-1-5-CloudClear**.

AIRS TLSCF Data System (TDS)

Catalog Access



- Set up UNIX environments upon log in:
 - setenv PATH $\${PATH}:/opt/packages/dom/bin$
 - setenv LD_LIBRARY_PATH \ $\${LD_LIBRARY_PATH}:/opt/packages/dom/lib$
 - setenv CATALOG_SERVER *airs-tlscf*

- Interfaces to access the catalog:
 - GUI-based catalog navigator:
 - *catnav*
 - Command line interfaces:
 - *cat_getfile*
 - *dom_getfile*

AIRS TLSCF Data System (TDS)

Catalog Access – The Catalog Navigator



DOM Catalog Navigator

File Filters Query Modes Servers Windows Help

Back Reload Home Server Path Finder Browse Query Super Browse Super Query

Catalog Collections on Server: airs_main_dev1_dom

```

  graph LR
    A[airs(5)] --- B[airs_input(3)]
    A --- C[airs_gdaac(3)]
    A --- D[airs_tlscf(3)]
    B --- B1[airs_input_1998]
    B --- B2[airs_input_2000]
    B --- B3[airs_input_2001]
    B1 --- B1a[airs_input_sep_1998]
    B2 --- B2a[airs_input_dec_2000]
    B3 --- B3a[airs_input_jan_2001]
    B3 --- B3b[airs_input_feb_2001]
    B3 --- B3c[airs_input_mar_2001]
    C --- C1[airs_gdaac_1998(1)]
    C --- C2[airs_gdaac_2000(1)]
    C --- C3[airs_gdaac_2001(3)]
    C1 --- C1a[airs_gdaac_sep_1998]
    C2 --- C2a[airs_gdaac_dec_2000]
    C3 --- C3a[airs_gdaac_jan_2001]
    C3 --- C3b[airs_gdaac_feb_2001]
    C3 --- C3c[airs_gdaac_mar_2001]
    D --- D1[airs_tlscf_1998(1)]
    D --- D2[airs_tlscf_2000(1)]
    D --- D3[airs_tlscf_2001(3)]
    D1 --- D1a[airs_tlscf_sep_1998]
    D2 --- D2a[airs_tlscf_dec_2000]
    D3 --- D3a[airs_tlscf_jan_2001]
    D3 --- D3b[airs_tlscf_feb_2001]
    D3 --- D3c[airs_tlscf_mar_2001]
  
```

Full tree – All Types



Catalog Access – The Command Line Interfaces

- `dom_getfile` is a wrapper program that wraps around the `cat_getfile` command.
- Similarity:
 - Same command line options
- Difference:
 - *dom_getfile* returns both data and metadata files in search result
 - *cat_getfile* returns only data file in search result



Catalog Access - The Command Line Options

**cat_getfile -r airs -t FileType -x MaxResultEntries [-cl-l]
-W'MetadataKeyword = "value"'**

- -r airs: specifies the root of a subtree where the search will start
- -t FileType: FileType specifies name of a TDS internal data file type that the user is interested in searching for in the query. For example: L1A_AMSU_T, L1A_HSB_T, Any_L1A_T, etc.

NOTE: All TDS-supported file types are documented in the working document entitled *TDS Data Hierarchy*.

- -x MaxResultEntries: MaxResultEntries is an integer that specifies the maximum number of files to be returned in the result file set. It provides a safety net for queries that may accidentally produce an excessive result set.
- [-cl-l]: These options are optional. If none is specified, the default is that a file list will be returned as the search result. Such file list that contains one file entry per line could then be used as input for other application program.



Catalog Access -The Command Line Options (cont.)

**cat_getfile -r airs -t FileType -x MaxResultEntries [-cl-l]
-W'MetadataKeyword = "value"'**

- When the `-c` option is specified, the search result will contain files that are physically copied over to the data user's current (local) working directory.
- When the `-l` option is specified, the search result will contain a set of symbolic links created in the data user's current (local) working directory.
- `-W'MetadataKeyword = "value"'`: the `-W` option allows data user to enter additional search criteria by specifying value for a predefined metadata keyword. In addition, multiple `-W` options can be specified in a query. In that case, all the metadata keywords will be joined with the logical "AND" operator.

NOTE: All supported metadata keywords/parameters are documented in the working document entitled *TDS Data Hierarchy*.

AIRS TLSCF Data System (TDS)

Catalog Access - Examples



- *cat_getfile -r airs -t L1A_AIRS_T -x50 -l *
-W'CollectionType="tlscf"' -W'DOMContainerDate="1998-09-
13"'
- *cat_getfile -r airs -t Any_L2_T *
-W 'CollectionType = "tlscf" AND DOMContainerDate >= 2001-01-
01 AND DOMContainerDate <= 2001-01-31 AND NumClearMW
>= 10'
- *cat_getfile -r airs_tlscf_jan_2001 -t Any_L2_T *
-W 'NumClearMW >= 10'
- *cat_getfile -r airs -t L1B_AIRS_Rad_T -l *
-W 'CollectionType = "gdaac" AND LatGranuleCen <= 5 AND
AutomaticQualityFlag = "Suspect"'